Downstream-Water-Level Control Test Results on the WM Lateral Canal

A. J. Clemmens, M.ASCE[1]; and R. J. Strand[2]

Abstract: On steep canals, distant downstream-water-level control can be challenging. The Software for Automated Canal Management was developed, in part, to test various distant downstream water-level controllers. It was implemented on the WM canal of the Maricopa Stanfield Irrigation and Drainage District, Stanfield, Ariz. to compare the performance of various controllers. In 2004, Clemmens and Schuurmans used optimization to determine the coefficients for a variety of controllers. These controllers vary in their complexity from a series of simple, single-input-single-output, proportional-integral controllers to a fully centralized, multiple-input-multiple-output, optimal controller. The controller design also varies regarding which pools are under downstream, or upstream, control and according to the conditions (e.g., flow rate) assumed for controller design. These controllers were tested under actual operating conditions and with unscheduled disturbances. The results of these tests are presented in this paper.

DOI: 10.1061/(ASCE)IR.1943-4774.0000079

CE Database subject headings: Irrigation districts; Canals; Automation; Computer software; Control systems; Water levels; Arizona.

Author keywords: Irrigation districts; Canals; Automation; Computer software; Control systems; Water levels.

Introduction

Canal automation is one of the many tools available for improving the operation of irrigation distribution systems. In a companion paper, Clemmens and Strand (2010) describe Software for Automated Canal Management, or SacMan, which can be used to implement a variety of canal automation methods. SacMan works in parallel with the commercial supervisory control and data acquisition (SCADA) software, and is able to read information from the SCADA database, and to cause the SCADA system to send information to local sites, for example, to change a gate position. The details of this interface are discussed in Clemmens and Strand (2010).

SacMan includes several automatic control features, including upstream control of water levels, distant downstream control of water levels, and flow control at canal head gates and check structures, if appropriate. SacMan provides flexibility in how the controls for a canal are configured, which will be demonstrated herein. SacMan also allows known delivery changes to be routed through the canal, but the focus of this paper is strictly feedback control. Feedforward control of known delivery changes will be discussed in a companion paper (Clemmens et al. 2010).

The software was tested on the WM lateral canal at the Maricopa Stanfield Irrigation and Drainage District (MSIDD), Stanfield, Ariz. for a period of 30 days from July 14–August 13, 2004. During that time, the canal was operational and delivering water to users, which constrained the severity of tests that could be conducted. Various water-level controllers were tested under both prescheduled and unscheduled flow changes. The purpose of this paper is to present the results of these tests and to discuss the advantages and disadvantages of various control strategies.

Downstream-Water-Level Control

Malaterre et al. (1998) provided a classification of control algorithms or methods. The approach presented here uses distant downstream water levels as the controlled variable and change in discharge at the upstream end of the pool as the control action. The design technique is the linear quadratic regulator (LQR). Ruiz-Carmona et al. (1998) described different methods for formulating canal control algorithms, including LQR, and show how they can be related to the classical proportional-integral (PI) control. Details of the LQR method used here are presented in Clemmens and Schuurmans (2004) and is summarized below for clarity.

With this implementation of LQR, we use the state-feedback control with a control law of the form

$$\mathbf{u}(k) = -K\mathbf{x}(k) \tag{1}$$

where $\mathbf{u}(k)$=vector of control actions at time k (one element of the vector for each control structure or gate); K=controller gain matrix; and $\mathbf{x}(k)$=vector of states at time k. Here the control actions are the changes in gate flow rates. A separate flow controller is used to adjust the gate position to provide the correct flow rate, which provides a master-slave control scenario.

A linear model is used to describe the change in downstream water level (controlled variable) as a function of a change in the flow rate at the upstream or downstream check structure (control action). We chose to use the integrator-delay (ID) model as the

[1]Center Director, Water Management and Conservation Research Unit, U.S. Arid Land Agricultural Research Center, 21881 North Cardon Ln., Maricopa, AZ 85238. E-mail: bert.clemmens@ars.usda.gov

[2]Electrical Engineer, Water Management and Conservation Research Unit, U.S. Arid Land Agricultural Research Center, 21881 North Cardon Ln., Maricopa, AZ 85238. E-mail: bob.strand@ars.usda.gov

Note. This manuscript was submitted on October 27, 2008; approved on February 23, 2009; published online on February 25, 2009. Discussion period open until December 1, 2010; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Irrigation and Drainage Engineering***, Vol. 136, No. 7, July 1, 2010. ©ASCE, ISSN 0733-9437/2010/7-460–469/$25.00.

linear process model (Schuurmans et al. 1999). The ID model can be expressed as

$$\frac{\Delta y(t)}{\Delta t} = \frac{1}{A_s}[-\Delta Q_{\text{out}}(t)] \quad t \leq \tau$$

$$\frac{\Delta y(t)}{\Delta t} = \frac{1}{A_s}[\Delta Q_{\text{in}}(t-\tau) - \Delta Q_{\text{out}}(t)] \quad t > \tau \tag{2}$$

where Δy=change of the downstream water level from its initial value; ΔQ_{in}=change in inflow at the upstream end of the pool; ΔQ_{out}=change of outflow at the downstream end of the pool; A_s =backwater surface area for the pool; τ=delay time for the pool; and t=time. Pool parameters A_s and τ can be obtained with a step test where the inflow rate at the upstream end of the canal pool is increased suddenly and the water level at the downstream end is observed, with the flow rate at the downstream end held constant.

Values of the gain matrix K are determined by minimizing the penalty function J

$$J = \sum_{k=0}^{\infty} \mathbf{e}(k)^T Q \mathbf{e}(k) + \mathbf{u}(k)^T R \mathbf{u}(k) \tag{3}$$

where $\mathbf{e}(k)$=vector of water-level errors at time k; Q=penalty function for water-level errors (usually an identity matrix); and R=penalty function for control actions (only main diagonal elements are nonzero). The matrices Q and R are discussed in more detail in the section on tuning. The solution of K is subject to the dynamic characteristics of the physical system, as described by the state-transition equations which are developed from the application of Eq. (2) to each pool, where for each pool i, $e_i(k)=y_i(k)-y_{\text{spi}}$, and y_{spi}=water-level setpoint. Note that for discrete incremental form, the state vector $\mathbf{x}(k)$ includes changes in water-level errors $\Delta\mathbf{e}(k)=\mathbf{e}(k)-\mathbf{e}(k-1)$, prior water-level errors $\mathbf{e}(k-1)$, and some prior control actions $\mathbf{u}(k-1)$, $\mathbf{u}(k-2)$, etc., to account for the time delay in each pool. Further details can be found in Clemmens and Schuurmans (2004).

Standard control engineering solutions are available for computing the gain matrix K that minimizes J, subject to the state-transition equations (Schuurmans 1997). The result is a multiple-input multiple-output (MIMO) PI controller where all water-level errors (and some prior changes in structure flow rates) influence the recommended changes to all structure flow rates $\mathbf{u}(k)$. Clemmens and Schuurmans (2004) found that they could set certain elements in K to 0 and solve for the remaining elements with optimization (*MathWorks; Matlab user guide* 2003). At one extreme, the solution for K provides a series of single-input single-output (SISO) PI controllers, one for each pool. Unfortunately, prior studies have suggested that a series of simple SISO PI controllers do not function as well as MIMO PI controllers (Schuurmans 1997; Clemmens and Schuurmans 2004; Clemmens and Wahlin 2004). If a disturbance occurs in the last pool downstream, with a SISO controller, control actions will be sent only to the next gate upstream, which causes a similar disturbance in the next pool, and this disturbance continues to cascade upstream. Thus, every pool must be disturbed before a control action will take place at the head of the canal. Schuurmans (1992) recommended an upstream decoupler to pass the information from a given pool to the upstream pools and a downstream decoupler to keep disturbances from traveling downstream. Here, flow control at structures effectively accomplishes this downstream decoupling. Deltour and Sanfillipo (1998) refer to this upstream decoupling as coordination, where PI signals from one pool are sent to all structures upstream. They also include a Smith predictor to account for the pool time delay. The optimization method proposed by Clemmens and Schuurmans (2004) allows intermediate controllers to be designed, to provide coordination among these simple PI controllers, and account for pool time delays by considering prior control actions.

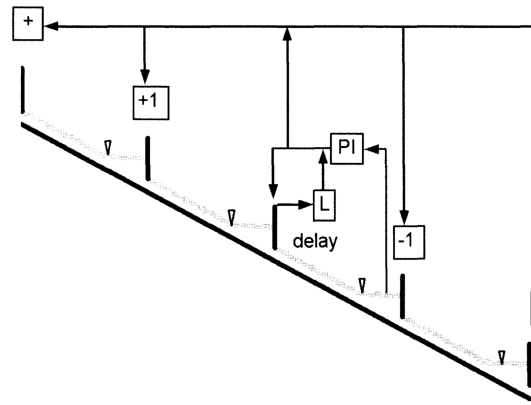

Fig. 1. Canal profile showing components of downstream water-level feedback control methods

For a single canal pool, Eq. (4) shows the various PI control elements for Eq. (1). It determines the change in upstream flow rate Δu at time k based on the proportional (P) constant K_p times the change in water-level error $\Delta e(k)$, and the integral (I) constant K_I times the previous water-level error $e(k-1)$. The two terms in the middle account for the delay or lag (L) time in the pool, by recognizing that an upstream flow change up to two time steps ago could still be influencing the water level now. The number of delay terms in Eq. (4) is simply the pool delay time divided by the control time step, rounded to the next highest integer

$$\Delta u(k) = K_P \Delta e(k) + K_{-2L}\Delta u(k-2) + K_{-1L}\Delta u(k-1) + K_I e(k-1) \tag{4}$$

Clemmens and Schuurmans (2004) developed a notational system to describe various control configurations. If Eq. (4) were applied to each pool independently, it would represent a PIL controller, where L indicates that the delay or lag time is considered. It is a PI controller if the middle two delay terms were omitted. A plus sign is used to indicate that control signals are sent to additional gates upstream; +1 for one additional gate upstream, and + for all upstream gates. A minus sign is used to indicate that control signals are sent to downstream gates; -1 for one additional gate downstream, and $-$ for all gates downstream. This is shown schematically in Fig. 1 where the water-level errors in the third pool are sent to the gate immediately upstream (PIL) and to one or more additional gates upstream and downstream. In Fig. 1, only the response from one water level is included, but the controller would respond similarly to each water level.

With this naming convention, PIL_-^+ would represent a PI controller that accounts for lag time (L), by taking into account prior control actions, and sends control signals to all upstream (+) and all downstream gates ($-$), or a full gain matrix. This is also referred to as a fully centralized PI controller. This is the controller that results from the classical LQR solution with all nonzero gain matrix elements. A PI$_{-1}^{+1}$ controller would represent a PI controller that does not consider prior control actions (no L) and sends control signals to one additional gate upstream and one additional gate downstream. The control scheme of Deltour and Sanfillipo (1998), in this notation, would be $\mathrm{PI}L^{+}$, although their implementation is different.

Clemmens and Wahlin (2004) examined the response of the following controllers on the ASCE Test Canal 1 (Clemmens et al. 1998); $\mathrm{PI}L_{-}^{+}$, $\mathrm{PI}L_{-1}^{+1}$, PI_{-1}^{+}, $\mathrm{PI}L^{+}$, PI_{-1}^{+1}, PI^{+}, $\mathrm{PI}L$, and PI. This order represents the expected quality of the performance based on the optimization results [i.e., value of J in Eq. (3)]. As one would expect, the fully centralized controller performed much better than a series of simple PI controllers. Simple gate position based PI controllers, with and without decoupling, were also evaluated. These gate-based controllers performed similarly to the flow-rate-based PI and PI^{+} controllers, to which they correspond. However, the slightly more complex flow-rate-based controllers performed much better, as expected from the optimization results. Exceptions include the $\mathrm{PI}L^{+}$ controller, which performed worse than the PI_{-1}^{+1} controller for the first test and worse than all the controllers on the second, more drastic test, where it went unstable. Also, the PIL controller performed slightly better than the PI^{+} controller, even though optimization results would have suggested otherwise.

Of interest is that in examining the gain matrix for the fully centralized controller, many of the coefficients that would send control actions downstream were very close to zero, except for the gate immediately downstream from the water level being observed. Because Eq. (3) represents quadratic criteria, the controller wants to spread the error out so that any one pool will not have a large deviation. The flow controller keeps the disturbance from being transmitted downstream, as does the downstream decoupling of Schuurmans (1992). Solution of this fully centralized controller shows that it wants to soften this downstream decoupling and allow some of this disturbance to travel downstream. This is counter to the notion of moving all of the disturbance to the upstream end. It was also surprising that the $\mathrm{PI}L^{+}$ and PI^{+} controllers performed more poorly than expected, with one case where the control became unstable. These results also suggest that perhaps allowing some of the disturbance to travel downstream helps with the overall stability.

Tuning is a major issue for any control design. Clemmens and Wahlin (2004) determined a single tuning constant to weigh the relative importance of water-level errors and structure flow changes, reflected by weights Q and R in Eq. (3), where only the diagonal elements are given values. Values in Q were set to 1, suggesting an equal penalty for water-level errors among pools. Values in R were weighted by the relative capacity squared, such that, for example, a 1-m^3/s change in a pool with a capacity of 10 m^3/s would have the same impact as a change of 0.5-m^3/s change in a pool with 5 m^3/s capacity. Then the single tuning

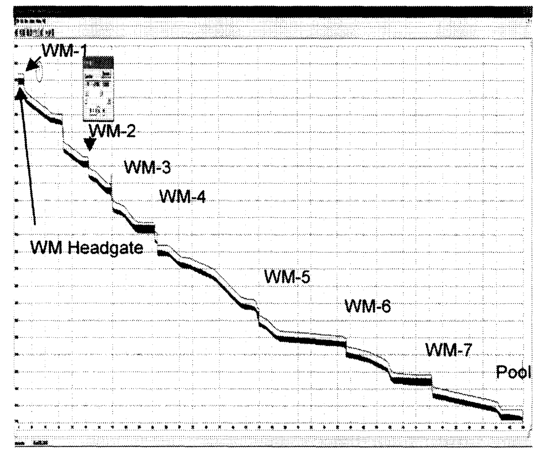

Fig. 2. Profile of the WM canal at 50% capacity

value R_1 or the first element of R reflected the relative importance of water-level errors in the first pool with flow-rate changes at the head gate. Here, penalties on the flow-rate changes serve to dampen the controller so that it does not over react and oscillate. Clemmens and Wahlin (2004) used simulation studies to determine an appropriate tuning factor. They used $R_1=20$ $(m/m^3/s)$ for ASCE Test Canal 1. Results were not highly sensitive to this value, with a value of 10 giving very similar results, although less damped, and a value of 1 giving very aggressive control, resulting in significant water-level oscillations. Wahlin and Clemmens (2006) found that $R_1=1$ gave good result for the Salt River Project Canal System. The Salt River Project canals are much larger and flatter than the ASCE Test Canal 1 (Clemmens et al. 1998), which could be the reason why smaller values of R_1 seemed appropriate. Currently, experience is required to determine appropriate values for R_1.

WM Canal Testing

The WM canal has a head-gate capacity of 2.5 m^3/s and is relatively steep with small backwater pools, particularly at the upstream end. The canal is 9.5 km long and drops almost 40 m. A profile is shown in Fig. 2. Pool properties were determined with unsteady-flow simulation of step tests in Sobek (*Sobek; manual and technical reference* 2000) at different flow rates: 80% of capacity in each pool, 40% capacity in each pool, and at a flow of 12% of head-gate capacity in all pools, as if only one delivery (of

Table 1. ID Model Parameters Used for Controller Design ($n=0.014$)

	Flow at head (%)	Pool							
		1	2	3	4	5	6	7	8
Capacity (m^3/s)	100	2.83	2.55	2.41	2.27	1.84	1.59	1.13	0.85
A_s (m^2)	80	343	450	240	1622	191	878	1500	1132
A_s (m^2)	40	379	600	493	1621	240	1385	1385	1319
A_s (m^2)	12	397	653	503	1630	100	1663	1473	1300
τ (min)	80	0	4.8	1.3	1.8	12.6	9.0	7.2	12.6
τ (min)	40	0	6.0	1.5	1.2	13.8	10.8	9.6	15.9
τ (min)	12	0	8.9	2.0	2.7	15.3	13.2	10.6	13.3
$p2d$ (min)	40	0	10	3	3	13.8	10.8	9.6	15.9

0.3 m^3/s) were being passed through the entire canal. The canal pool properties are shown in Table 1.

Matlab (*MathWorks; Matlab user guide* 2003) was used to design a number of different controllers based on these properties with Eqs. (1)–(3). Because the WM canal formed the basis for the ASCE Test Canal 1, we used the tuning constant obtained by Clemmens and Wahlin (2004) or $R_1 = 20$. The feedback control time step was 10 min for all controllers and the flow control time step was 2 min. Testing with unsteady-flow simulation suggested that this would provide reasonable results (e.g., see Clemmens et al. 2002).

During 2004, the WM canal was under automatic control for a period of 30 days. In this paper, we present results for feedback control tests, where a disturbance was created in the canal and the feedback controller was forced to react to the disturbance and bring all water levels back to their set points. A total of 19 different downstream controllers were tested, of which only a few can be presented here. Other papers will deal with routing deliveries to users, upstream control, and start-up and emergency control issues. During initial testing, electrical storms caused wells that had been pumping groundwater into the canal to shut off, thus creating a disturbance. An example is given in Clemmens and Strand (2010). This proved to be a good way to test different controllers under more-or-less the same conditions. In some cases we actually turned wells on or off, and did this as a known scheduled change or as an unknown unscheduled change, where the controller did not know about the change even after it happened. However, we found it easier to fake a well outage by routing a negative flow change down to the location of the well and then not changing anything, thus causing a flow mismatch (unscheduled well outage test). We could do this without coordinating turning the well on or off with the district staff.

SacMan is flexible such that one or more pools can be skipped by the downstream controller. In such cases, the check structure is placed under upstream level control, and the delay time for the next pool downstream accounts for the delay in both pools. For this canal, we found it more reliable to put WM-5 under upstream level control since it has a very small pool and a relatively long delay.

The performance of the various controllers was evaluated with the criteria described by Clemmens et al. (1998) with a minor modification. These include the maximum absolute error (MAE), which is expressed as a percentage of the set-point water depth; integral of absolute magnitude of error (IAE), which is essentially the absolute value of the average error as a percentage of the set-point water depth; and the integrated absolute discharge change (IAQ), which is as expressed here as the absolute value of the average discharge change excluding the difference between initial and final discharges. To get this value, the expression for IAQ in Clemmens et al. (1998) is divided by the number of discharge changes n. This allows tests with different durations to be compared.

Results

Downstream Control for Setting Canal Inflow

During the early morning of July 17, 2004, the canal was being operated under upstream level control in all pools. The canal inflow was nearly constant. The canal head gate was under automatic flow-rate control, but this was strictly based on gate position since it did not have information on water levels. Fig. 3 shows the water levels from midnight (00:00) to 06:00. During the first few hours, the water levels appear to all be relatively constant. Upstream water-level control tests will be discussed in more detail in a companion paper (Clemmens et al. 2010). The important point here is that any flow-rate errors are moved to the downstream end of the canal, which for this test was pool WM-7, since water was not being delivered beyond this pool. Note that the water level there is gradually rising, suggesting that the canal inflow was too large or that one of the turnouts was not taking enough water, which could result from debris clogging the turnout gates. The dashed lines in this figure represent the water-level setpoint used by the controller.

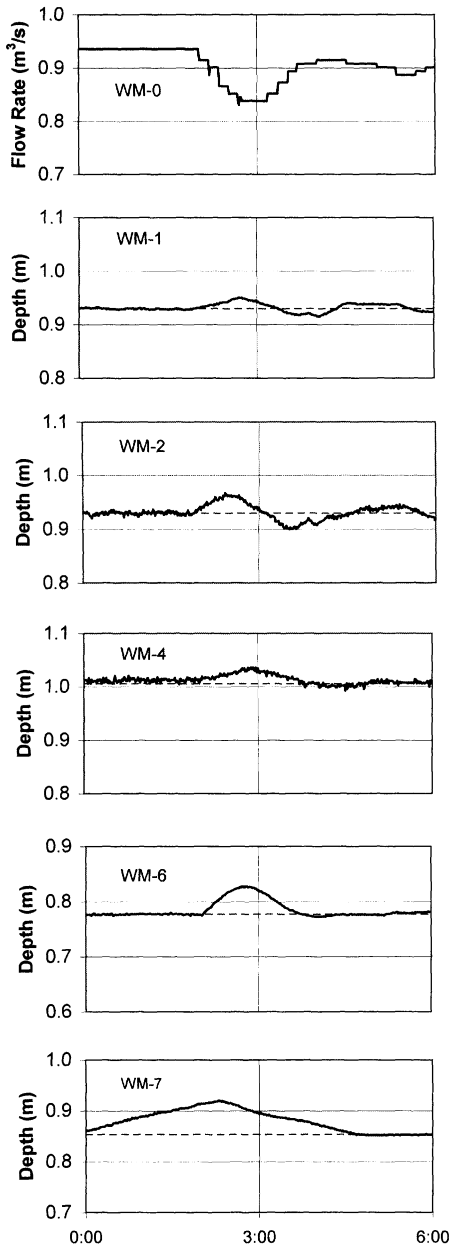

Fig. 3. WM canal inflow and water-level response with PI^{+}_{-1} controller (80%), July 17, 2004. Initial inflow did not match the outflow. (WM-5 under upstream control, not shown).

At 01:50, the control was changed to downstream water-level control with a PI^{+}_{-1} controller designed at 80% capacity with WM-5 under upstream level control. This controller had been used as the default controller from previous tests. Note that the

Table 2. Performance Parameters for Controllers Tested

Testor figure number	Controller	Date-time	IAQ/n (m^3/s)	Pool 1 IAE (%)	Pool 1 MAE (%)	Pool 2 IAE (%)	Pool 2 MAE (%)	Pool 3 IAE (%)	Pool 3 MAE (%)	Pool 4 IAE (%)	Pool 4 MAE (%)	Pool 5 IAE (%)	Pool 5 MAE (%)	Pool 6 IAE (%)	Pool 6 MAE (%)	Pool 7 IAE (%)	Pool 7 MAE (%)
3	PI$^{+}_{-1}$ (80)	July 17 02:00–04:00	0.0010	0.8	1.5	0.8	2.0	0.6	1.6	0.3	1.3	U/S	U/S	0.3	0.7	0.5	2.5
4	PI (40)	July 22 00:46–09:00	0.0081	2.9	7.1	3.3	7.5	0.9	2.5	0.5	1.8	U/S	U/S	0.5	1.2	0.4	0.8
4	PI$^{+}_{-1}$ (40)	July 22 10:00–12:00	0.0034	1.4	2.7	1.0	2.4	0.3	1.0	0.3	1.3	U/S	U/S	0.2	0.6	0.2	0.4
5a	PIL^{+}_{-} (40) well off	August 1–2 20:18–00:18	0.0079	1.6	6.4	2.0	10.7	4.0	26.1	2.5	7.1	1.1	3.6	1.5	3.7	2.4	3.0
5b	PI$^{+}_{-1}$ (40) well off	August 2 17:48–21:48	0.0076	0.7	2.4	1.8	11.8	4.0	23.9	3.1	11.0	0.8	2.7	0.7	2.3	0.4	0.9
A	PIL^{+}_{-} (40) fake well off	July 31 02:23–06:23	0.0028	1.9	5.6	1.4	5.2	3.4	24.6	2.9	6.4	1.2	3.7	2.1	5.1	No flow	No flow
B	PI$^{+}_{-1}$ (40) fake well off	July 30–31 20:52–00:52	0.0044	3.1	9.0	2.6	11.3	3.8	24.8	2.8	6.8	1.0	2.6	0.7	1.8	No flow	No flow
6	PI^{+} (40) fake well off	August 11 04:47–08:57	0.0257	2.1	5.4	6.9	24.0	4.9	23.9	1.7	6.4	1.4	4.8	1.0	5.2	0.6	3.3
C	PI$^{+}_{-1}$ (40) p2d fake well off	August 4 02:24–06:24	0.0065	1.9	9.5	3.1	14.8	4.1	20.0	3.3	10.9	1.1	5.1	0.6	2.3	0.3	1.0
7	PI$^{+}_{-1}$ (40) fake well off	July 27 13:07–17:07	0.0065	U/S	U/S	3.2	15.4	4.3	24.3	2.1	6.1	U/S	U/S	1.0	3.3	0.5	2.2
D	PI$^{+}_{-1}$ (40) fake well off	July 24 02:21–06:21	0.0075	U/S	U/S	3.9	11.8	U/S	U/S	3.9	12.4	U/S	U/S	1.6	5.0	0.5	1.5
8	PIL^{+}_{-1} (40) fake well off	July 28 19:53–23:53	0.0189	U/S	U/S	5.3	12.3	4.8	22.8	2.0	6.5	U/S	U/S	1.0	2.9	0.6	2.2
E	PIL^{+}_{-} (40) fake well off	July 29 00:17–04:17	0.0094	U/S	U/S	3.6	13.1	3.5	22.5	1.6	4.7	U/S	U/S	1.2	4.1	0.7	2.2

Note: U/S=under upstream control.

controller immediately started to reduce the flow at the head gate. Each check gate, except WM-5, was also closed to release less water downstream, resulting in a rise in the water levels in the pools upstream. The controller continued to make adjustments until all water levels were back to the set point by about 04:00 or 05:00. Note that the canal inflow oscillated a little. The initial big drop in the flow was necessary to remove the excess volume from the canal. By about 04:00 it was very close to the desired flow, but continued with fine adjustments since all water levels were not at their set points. Even by 06:00, small deviations can be seen in the first few pools, even though the downstream pools appear to remain at the water-level set points. Results after 06:00 are not shown since new flow changes were made on the canal.

Controller performance is shown in Table 2. Results are shown between 02:00 and 04:00 to show more long-term behavior since most initial deviation had been removed. The MAE value for Pool 7 still reflects part of the initial error, while MAE values for Pools 1 and 2 are the result of oscillation. The controller demonstrated in Fig. 3 stabilized the canal fairly well (IAE well within 1%, or less than 7-mm average deviation) even though the controller was designed for a flow rate at 80% capacity (2.27 m^3/s at the head gate), while the canal inflow was only about 35% of capacity (0.9 m^3/s). This suggests that this controller is fairly robust to discharge changes.

Limitations of Simple PI Controllers

Fig. 4 shows the results from the morning of July 22, 2004. The prior evening, some tests had been run and the canal water levels and inflow were reasonably stable. At 00:46, the controller was changed to a PI controller designed at 40% capacity with WM-5 under upstream level control, which is equivalent to a series of simple PI controllers. The vertical lines in these drawings represent communication gaps where water levels were not recorded. This occurred for various reasons but such gaps did not affect control. The change in controller was accomplished without restarting the controller, by simply changing the gain matrix K and thus avoiding any start-up issues.

Here one can see that there were initially a few small deviations from the set points, generally less than 1 cm. The PI controller started to react to these small deviations. One can see that the errors in water level began to build over time. No significant external disturbance occurred during this test. These cycles were caused by the controller. It is clear that this controller was cycling and the disturbances appeared to be growing. At 09:00, the controller was switched to a PI$^{+}_{-1}$ controller, designed under identical tuning conditions (40% capacity, WM-5 under upstream level control). A water order change was requested for an outlet from Pool WM-4 at 10:00. This was scheduled by SacMan Order and implemented automatically. The step change at the head gate can be seen at about 9:40, since it takes about 20 min for water to travel to WM-4 based on feedfoward control (Bautista and Clemmens 2005). Even though this controller was dealing with a disturbance traveling through the canal, its water-level response is much improved over the series of simple PI controller. The results of these two controllers are shown in Table 2 (rows for Fig. 4). The values for the PI$^{+}_{-1}$ controller are shown after the delivery

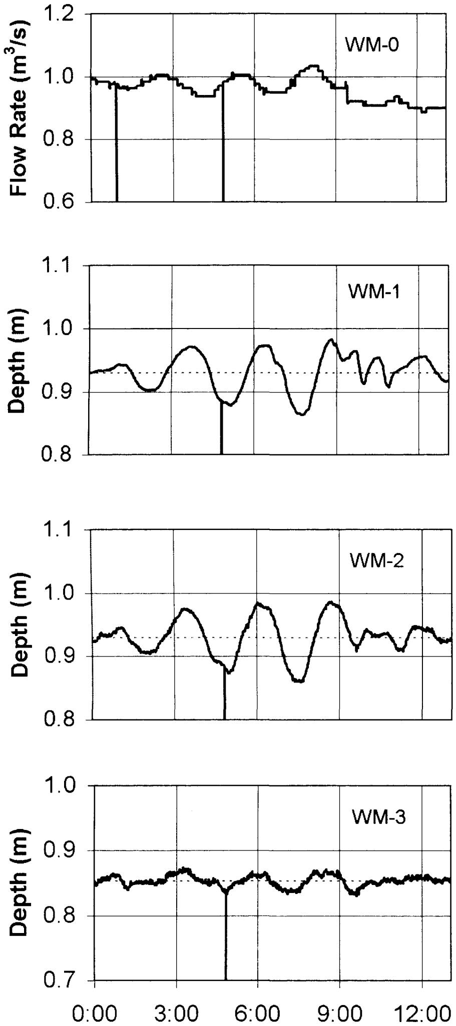

Fig. 4. WM canal inflow and water-level response with PI controller (40%), July 22, 2004. No disturbance (WM-5 under upstream control, downstream pools are not shown).

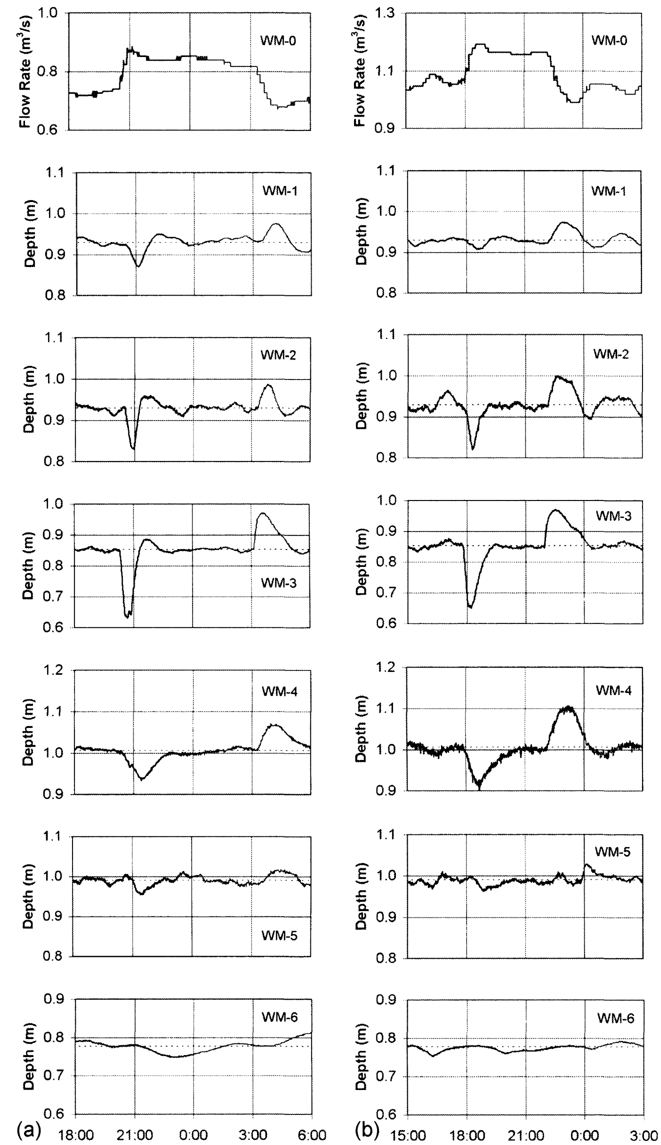

Fig. 5. WM canal inflow and water-level response to unscheduled well outage test with PIL^+_- and PI$^+_{-1}$ controllers (40%), August 1–2 and 2–3, 2004

change was made at WM-4. All the values for the PI controller are about double the values for the PI$^+_{-1}$ controller. Because of this poor performance, we limited the testing of these simple PI controllers since we were delivering water to users during this testing.

Advantages of More Complex Controllers

Fig. 5 shows the results of two sets of tests on the evenings of August 1–2 and August 2–3, 2004, where we turned off the well at WM-3 unscheduled, and then turned it back on several hours later, also unscheduled. Two different controllers were tested PIL^+_- (August 1–2) and PI$^+_{-1}$ (August 2–3), both designed at 40% capacity for all pools. The wells were turned off at 20:18 August 1 and 17:48 August 2. The wells were turned back on at 03:08 August 2 and 21:58 August 2, for the two controllers, respectively. The PIL^+_- controller [Fig. 5(a)] sends control signals to all upstream and downstream gates and considers the impact of prior control actions (fully centralized PI controller), while the PI$^+_{-1}$ controller [Fig. 5(b)] sends control signals to all gates upstream and one gate downstream. The mismatch in inflow and outflow caused the water level in Pool WM-3 to drop when the well was turned off and rise when the well was turned on. The results for these two controllers are shown in Table 2 [Figs. 5(a and b), respectively]. The results are shown for a 4-h stabilization period. There was essentially no difference in IAQ, suggesting that overall, both controllers responded similarly. Note the very large MAE in Pool 3 caused by the well outage. Most of the disturbance was spread to Pools 2 and 4 as indicated by the large MAE values there. The PIL^+_- controller had smaller MAE values in Pools 2, and 4, and larger MAE values elsewhere, as it tried to spread the error out among all pools. However, this spreading out of the error resulting in larger IAE values, particularly at the lower end of the canal, as shown by the larger water-level deviation at WM-6. For comparison to other controllers described below, the results of an unscheduled well outage at WM-3 are shown in Table 2 (Tests A and B). Here, a flow change was routed from the head gate to Pool 3, and then no change was made there, resulting in a flow mismatch. In this case, the PIL^+_- controller (Test A) had a much smaller IAQ, and it had smaller IAE and MAE values for Pools 1–4, and larger values for Pools 5 and 6. (No water was flowing in Pool 7.) In this case, the control was improved by spreading the error out among pools. Both controllers were able to stabilize the canal relatively quick, i.e., within a few hours. There is no clear choice among these two controllers based on these two sets of results.

It should be noted that this unanticipated flow mismatch was only 0.1 m^3/s in a canal with a capacity of 2.5 m^3/s, or about 4% of design capacity at the head gate. Further, this change was made relatively close to the head gate. Based on delay times at 40% capacity, the delay to this pool is only 7.5 min, or about 13% of the total delay time for the canal. Thus, one can see that even this small flow mismatch, this close to the head gate, caused very large water-level deviations before the downstream feedback controller was able to correct the situation. This clearly demonstrates the importance of routing known flow changes through the canal, i.e., feedforward control actions, which will be discussed in a companion paper.

Response of Simple Coordination

The PI^+ controller represents the simplest form of coordination where all control signals are sent to all upstream gates. An unscheduled well outage test was run with a PI^+ controller designed with the same criteria (40% capacity, all pools). For this test, a negative flow change matching the gate discharge was routed to WM-3, with no change at WM-3. The results are shown in Fig. 6 and Table 2, where graphs for Pools 6–8 are not shown for brevity since water-level deviations are relatively minor. This test was started at roughly 05:00. With the previous controllers (Test A and B), the transients would have been mostly gone by 08:00 when the first scheduled flow changes occurred. The IAQ value for this test (0.0257 m^3/s) was 5 to 10 times greater that the IAQ values for Tests A and B (0.0028 and 0.0044 m^3/s, respectively), suggesting that the controller was oscillating, as shown in the WM-0 graph of Fig. 6. Unfortunately, scheduled flow changes starting at 06:45 at the head gate made more direct comparisons difficult. One can see the two changes at the head gate at about 06:45 and 10:45 in Fig. 6. The PI^+ controller should have had no deviations below Pool 3, since the flow rate there is not influenced by upstream levels. Yet because of the rapid drop in the water level at WM-3, the flow controller was always behind in its flow adjustments, thus unintentionally sending flow errors downstream. Because essentially all water levels were sent to all pools upstream, the water level at WM-2 dropped significantly more (MAE at 24.0%) than for other controllers (Tests A and B, MAE at 5.2 and 11.3%, respectively). In this case, it dropped as much as that of WM-3. More important, it is clear that the level in WM-2 had started a fairly large oscillation which is reflected in the oscillation in the canal inflow. The scheduled change at 06:45 had some influence on the water levels, but it did not trigger nor aggravate these oscillations. At those times, one sees only a minor bend in the overall feedback response. These oscillations continued until a different controller was implemented.

The poor response of the "PI^{+}" controller was a bit surprising, although it is consistent with the results of Clemmens and Wahlin (2004). We hypothesized that the delay times used for the controller design were a little short. However, the other two controllers shown in Fig. 6 used the same delay times. This suggests that the simpler controllers are more sensitive to the conditions used for controller tuning and thus are less robust. To determine whether the pool delay times were influencing these oscillations in WM-2, we tested controllers with longer delays in Pools WM-2 through WM-4 (Table 1). The same unscheduled well outage test was run with a PI^+_{-1} controller with the modified delay times. The

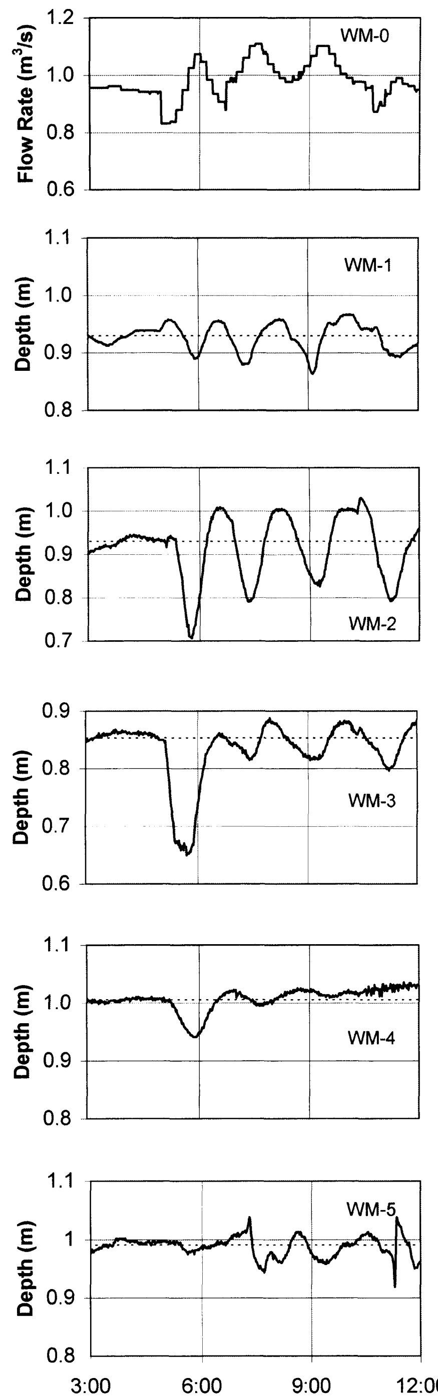

Fig. 6. WM canal inflow and water-level response to unscheduled well outage test with PI^+ controller (40%), August 11, 2004. (WM-6 to WM-8 not shown)

response for most indictors (Test C) is worse than that in the original controller (Test B), as shown in Table 2.

Removing Canal Pools from Downstream Feedback

As mentioned earlier, we had difficultly with Pool WM-5 when there were significant disturbances at the downstream end of the canal. So we chose to design controllers with WM-5 under upstream level control. In most cases, this had only a small effect on the response of water levels in other pools, as suggested by Fig. 3. Also, Pool WM-1 is very short, subject to wave action and tended

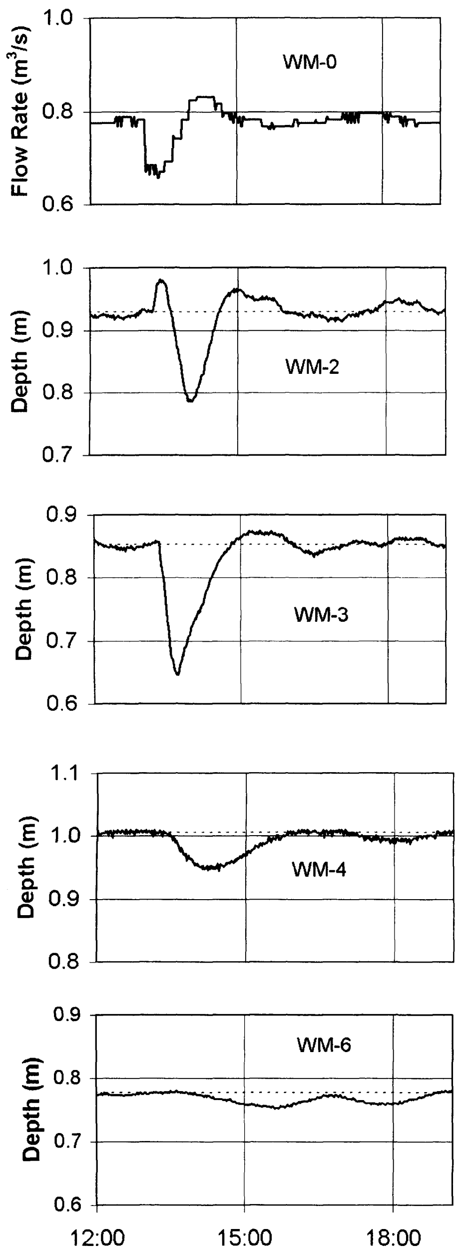

Fig. 7. WM canal inflow and water-level response to unscheduled well outage test with PI^{+}_{-1} controller (40%), July 27, 2004. (WM-1 and WM-5 under upstream level control)

to oscillate when Pool WM-2 oscillated. The water level at WM-1 was also a little sensitive to flow errors during feedforward routing (i.e., making flow changes at WM-0 and WM-1 that did not match). We thought that perhaps putting Pool WM-1 under upstream level control might overcome this problem and help avoid oscillations in Pool WM-2. Since Pool WM-1 did not add significantly to the delay in changes arriving at WM-2, it was thought that this would not delay control actions arriving there.

Fig. 7 shows the response of a PI^{+}_{-1} designed at 40% capacity with WM-1 and WM-5 under upstream level control. Those water levels are not shown since they were relatively constant. This test should be compared to the first half of Fig. 5(b). First, note that the timing of the flow change generated by the feedforward controller at WM-2 was off. (For Test B Table 2, this deviation actually occurred in pool WM-1.) WM-2 had a slightly larger deviation in Fig. 7 than in Fig. 5(b) (MAE at 15.4% versus 11.8%), perhaps because WM-1 was not there to help buffer volume changes. Pool WM-4 actually had a smaller deviation (MAE at 6.1% versus 11.0%) perhaps because it did not have to respond to the changes at WM-5. Pool WM-6 had a slightly larger change in water level (MAE at 1.0% versus 0.7%), which also might be attributed to the lack of downstream control at WM-5, but the effect is fairly small. Overall, the controller responded fairly well to the disturbance, even with two pools missing.

To further test this approach, we also conducted the unscheduled well outage test with Pools 1, 3, and 5 under upstream level control. As expected, the disturbance in Pool 3 was sent to Pool 4 by the upstream controller (Table 2, Test D). The IAQ for this controller was slightly higher (0.0075 rather than 0.0065 m^3/s). The MAE value was much higher in Pool 4, as expected (12.4% versus 6.1%). The other response parameters were similar.

Accounting for Prior Control Actions

Fig. 8 shows the results of a PIL^{+}_{-1} controller with the same conditions as in Fig. 7 (40%, no WM-1 or WM-5). The only difference between this controller and the one in Fig. 7 is that this controller takes prior control actions into account (L). The large oscillations and poor performance of this controller was very surprising (IAQ of 0.0189 m^3/s versus 0.0065 m^3/s). The performance of a PIL^{+}_{-} controller, with WM-1 and WM-5 under upstream level control, was also tested for comparison (Table 2, Test E). IAQ is slightly worse than that for PI^{+}_{-1}, Fig. 7 (0.0097 m^3/s versus 0.0065 m^3/s), and oscillations were minor. IAQ for the PIL^{+}_{-1} (Fig. 8) controller is twice than that for PIL^{+}_{-} (Test E), and the only difference is that control actions are only sent to one gate downstream rather than all gates. Since many of these coefficients are nearly 0, one would not expect this degradation in performance. Clemmens and Wahlin (2004) observed this poor performance for PIL^{+} but not for PIL^{+}_{-1}.

Discussion

Clemmens and Wahlin (2004) noted for ASCE Test Canal 1 that the PI^{+} and PIL^{+} controllers performed worse than expected. On the actual WM canal, these results suggest that the PI^{+} and PIL^{+}_{-1} controllers performed poorly or worse than expected. (We did not test a PIL^{+} controller.) These results were a little surprising. These controllers were all designed with the same performance criterion. It is well known that controllers with a lag time prediction can perform poorly if the timing is wrong. Yet past experience suggests that adding control of the additional gate downstream is usually enough to overcome such a problem. We suggest that small differences in actual and expected conditions have caused these controllers to perform poorly, suggesting that they are not very robust for this canal. Detailed studies on robustness were not performed. It is also possible that the optimization results somehow provided inconsistent results for these particular controllers, although this was not reflected in the value of objective function J.

Controller performance degrades when the actual conditions differ from the conditions for which the controller was designed. This may suggest the need to use different controllers under different operating conditions, the so-called gain scheduling. This is relatively straightforward to do with the SacMan software that

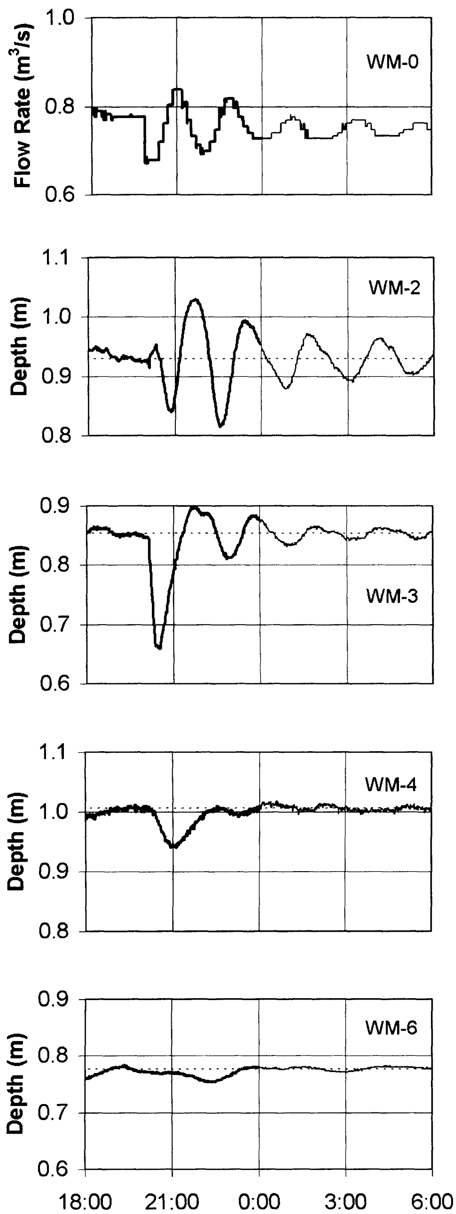

Fig. 8. WM canal inflow and water-level response to unscheduled well outage test with PIL^{+}_{-1} controller (40%), July 27–28, 2004. (WM-1 and WM-5 under upstream level control)

was used to test control of the WM canal, although such scheduling is not automatic.

The performance of any downstream water-level feedback controller is really limited by the physical characteristics of the canal. The WM canal is steep with little storage. This is more common with lateral canals, compared to larger main canals, which tend to run more on the contours. In any case, the WM canal is somewhat of an extreme case, also though we have experienced other canals with similar difficulties. A canal that has more backwater behind check structures should be able to handle larger flow mismatches than we were able to handle with the WM canal. In a companion paper, we discuss how to deal with large water-level fluctuations caused when flow mismatches exceed the canal's ability to adequately store short-term mismatches. When the canal slope is very shallow, reflection waves in the canal pools influence controllability. This is discussed in Schuurmans (1997) and will be the focus of future research.

Conclusions

This test on an actual canal under operating conditions showed the following:

- Upstream water-level controllers move disturbances to the downstream end of the canal where they can accumulate and/or spill.
- Downstream water-level controllers will adjust the canal inflow to balance canal outflow.
- SacMan was successful in being able to test a variety of downstream control approaches on the WM lateral canal at the MSIDD.
- The design approach used here developed a stable distant downstream water-level controller for this canal which functioned well even when flows were different than assumed for controller design.
- Test results showed that the performance of a series of simple PI downstream water-level controllers is far inferior to the performance of more sophisticated controllers for this canal.
- Experience with this canal suggests that the PI^{+}_{-1} controller provides better and more consistent performance overall. It appears that this controller has a good mix of performance and robustness.
- Simulating an unscheduled well outage was an effective way to test various controllers on this canal.
- The control performance was better and more robust when control actions were passed to at least one structure downstream, an action that partially spreads disturbances downstream, i.e., it softens the downstream decoupling.
- These tests demonstrate that some canal pools can be skipped by the downstream controller and put under upstream control. This may cause minor degradation in performance.
- Controllers that account for prior control actions in the controller gain matrix seemed to be less robust when actual conditions differed from that of the assumed during the controller design.

References

Bautista, E., and Clemmens, A. J. (2005). "Volume compensation method for routing irrigation canal demand changes." *J. Irrig. Drain. Eng.*, 131(6), 494–503.

Clemmens, A. J., Kacerek, T. F., Grawitz, B., and Schuurmans, W. (1998). "Test cases for canal control algorithms." *J. Irrig. Drain. Eng.*, 124(1), 23–30.

Clemmens, A. J., and Schuurmans, J. (2004). "Simple optimal downstream feedback canal controllers: Theory." *J. Irrig. Drain. Eng.*, 130(1), 26–34.

Clemmens, A. J., and Strand, R. J. (2010). "Application of software for automated canal management (SacMan) to the WM lateral canal." *J. Irrig. Drain. Eng.*, 136(7), 451–459.

Clemmens, A. J., Strand, R. J., and Bautista, E. (2010). "Routing demand changes to users on the WM lateral canal with SacMan." *J. Irrig. Drain. Eng.*, 136(7), 470–478.

Clemmens, A. J., Strand, R. J., Feuer, L., and Wahlin, B. T. (2002). "Canal automation system demonstration at MSIDD." *Proc., USCID/EWRI Conf.*, USCID, Denver, 497–506.

Clemmens, A. J., and Wahlin, B. T. (2004). "Simple optimal downstream feedback canal controllers: ASCE test case results." *J. Irrig. Drain. Eng.*, 130(1), 35–46.

Deltour, J.-L., and Sanfillipo, F. (1998). "Introduction of Smith predictor into dynamic regulation." *J. Irrig. Drain. Eng.*, 124(1), 47–52.

Malaterre, P. O., Rogers, D. C., and Schuurmans, J. (1998). "Classification of canal control algorithms." *J. Irrig. Drain. Eng.*, 124(1), 3–10.

MathWorks; Matlab user guide. (2003). The MathWorks, Inc., Natick, Mass.

Ruiz-Carmona, V., Clemmens, A. J., and Schuurmans, J. (1998). "Canal control algorithm formulations." *J. Irrig. Drain. Eng.*, 124(1), 31–39.

Schuurmans, J. (1992). "Controller design for a regional downstream controlled canal." *Rep. No. A668*, Laboratory for Measurement and Control, Delft Univ. of Technology, Delft, The Netherlands.

Schuurmans, J. (1997). "Control of water levels in open channels." Ph.D. dissertation, Delft Univ. of Technology, Delft, The Netherlands.

Schuurmans, J., Clemmens, A. J., Dijkstra, S., Ahmed, R. H., Bosgra, O. H., and Brouwer, R. (1999). "Modeling of irrigation and drainage canals for controller design." *J. Irrig. Drain. Eng.*, 125(6), 338–344.

Sobek; manual and technical reference. (2000). WL|Delft Hydraulics, Delft, The Netherlands.

Wahlin, B. T., and Clemmens, A. J. (2006). "Automatic downstream water-level feedback control of branching canal networks: Simulation results." *J. Irrig. Drain. Eng.*, 132(3), 208–219.